RESEARCH BRIEF

CONTEXT LEARNING IN OKAPI

AYSE GOKER
goker@bilkent.edu.tr

Bilkent University, Bilkent, Ankara, Turkey

A user who makes repeated use of a retrieval system may be assumed to have a context which is common to successive uses (even if the immediate need differs). An IR system which could make use of this context may be better able to match the specific need. A machine-learning approach to inferring the user's context is described, and the results of an evaluation experiment are given. There appears to be scope for IR systems to operate in this way.

INTRODUCTION

An adaptive systems approach which applies machine learning techniques to IR has been developed and tested on the Okapi retrieval system [1]. Various statistical or probabilistic methods have been used previously to improve system performance, but have focused on the user's search within a single online session. The main difference in the work described here is that it encompasses the idea of *learning* from one session to another. What is 'learned' in one session can be used to improve document ordering, for example, for the next session of a particular user.

The basis of this adaptive system is that a user has what is called a 'context' within which he or she forms a query, at any particular time. This notion of a context will be explained shortly, but it is based on an earlier finding that users do not present a random set of queries from one session to another, but that there is in fact a connection between their search topics [2]. This is precisely what opens up the possibility of learning, not only within the same session, but also from one query to another across sessions.

SCOPE AND MEANING OF CONTEXT

The term 'context', as it is referred to in this work, can be explained within the framework of the ASK (Anomalous State of Knowledge) model [3]. According to this model, there are three stages in an information retrieval situation. Firstly, a person finds himself or herself in a problematic situation for which his or her internal resources are inadequate. This results in an ASK, the second stage. In order to resolve this the person has recourse to an external resource for which a possible response is an information need, the third stage. This need results in a query which is then expressed in terms of a search statement. The 'problem

situation' described can be said to represent the context within which the query occurs and the aim of the context learner is to help resolve the ASK. As mentioned previously, users tend to repeat searches or conduct a series of closely related searches over a period. Although each search must be regarded as representing a different information need, they can all be assumed to have a common context. However, a document that is judged relevant to the need which prompted one search is not necessarily relevant to the next need – it is only within the same context.

Representation of context

A context C for query Q is a representation of a particular user's context at a particular point in time. Context C_1 for query Q_1 and C_2 for Q_2 are related, strongly or weakly. The aim is to find some way of deriving these contexts.The strong hypothesis would state that C_1 and C_2 are equal, whereas the weak would simply state that they are related since there may be a gradual shift in the user's information need. In practice a context can never be represented completely accurately; we can only hope to approximate it. Hence, the notation below refers to successive approximations of C, for application purposes, before and after a query.

Let us consider the current query to be Q_i, and the previous query, therefore, to be Q_{i-1}. Using ′ to denote a context *before* a query and ″ to denote it *after* a query, the context before Q_{i-1} is C_{i-1}' and the context after Q_{i-1} is C_{i-1}''. Likewise the context before the query Q_i is C_i'. Theoretically, C_{i-1}'' and C_i' are different, since it is possible for further information about the user or subjects of interest to emerge between queries. However, for the purpose of the context learner discussed here, they are assumed to be equal.

THE CONTEXT LEARNER

Contexts are formed after each query, for each user, in order to help document ordering for the next query. (Note that in the experiments described below, this document ordering was subsidiary to the standard Okapi probabilistic ranking: it was applied to blocks of documents with the same basic 'score'.) A context learner (CL) consists of four modules, each dealing with a different stage in the process of identifying terms to be included in a context, and determining how they are used to affect subsequent document ordering. Each module has various possible parameters.

The context learner modules

The first module specifies the set of relevant documents and how it should be obtained (e.g. what information is to be used and how to treat duplicate relevant records). A potential list of terms useful for the context is formed, and the second module divides this set into two, an active set (A) and a passive set (P). The criteria for forming these sets include statistical data about a term such as its weight, and the number of relevant documents containing it. Additional

parameters include a limit on the total number of active terms. The third module deals with merging sets of newly acquired terms with the previous context to form a new context representation. In doing so, a fourth module is used which [illegible] already in the context, to determine its current position. A similar algorithm is applied to terms once they are in a context. Terms in the active set are used for reordering documents, those in the passive set are put on hold. Terms shift from one set to another when the relevant criteria are met. Generally, the more frequently a term appears in a user's relevant documents, the more likely it is to remain in the active set. Its 'currency' (how recently it occurred in a relevant document) will also influence its position.

EVALUATION

Ideally, all possible parameters of each module and all possible combinations of modules should be tested and evaluated, and using one of the test collections would have provided one way to do this. However, the main drawback of test collections is that their queries are independent of one another, and thus unsuitable for context derivation. Hence the approach which was taken involved obtaining relevance judgements from actual frequent users of the system, for whom contexts had been formed. With such a user-oriented evaluation, however, it was not feasible (in terms of number of users, queries and required relevance judgements) to test for *all* permutations of context learner modules and parameters. Hence, a two-stage evaluation was carried out, consisting of two experiments to establish which modules/algorithms performed better in terms of relevance. The first stage used objective criteria to eliminate some versions from a large candidate set; the second focused on a more in-depth analysis of the remaining candidates, and compared them with the existing Okapi system.

Experimental methods

The evaluation experiment required the identification of a group of frequent Okapi users, so that a context could be derived for them based on their history of system usage. At the time of the experiment, Okapi had been available at City University for some time with a section of the Inspec database, so that it was possible to find regular users. Furthermore, as Okapi asks users to make relevance judgements on documents viewed during the search, the necessary data for the context learner was available. For simplicity, only a batch mode version of the program was implemented.

Each user was asked to provide new relevance judgements in relation to his or her last use of the system, i.e. the information need that prompted the last use. To make the tests as realistic as possible, the judgements were requested shortly after the search, and all those from a given user were made at one time. Each version of the context learner was given the historical data (up to but not including the last search), to derive a representation of the context immediately prior to the last search. Simulations were then run of how the last search would

have proceeded using that version. Judgements were obtained on the pooled output of all searches, a small number of documents being taken from the top of each ranked list in order to avoid overloading the users. In the first experiment, the searches were based on various versions of the context learner; in the second, a shortlist of context learners was compared to plain Okapi.

RESULTS

The results showed that users' contexts do change over time, although the rate of change is slow. Interestingly, it was found that a simple algorithm, using term frequencies in related documents and limiting the number of terms in the context, appeared to be more successful than more complex algorithms. Contexts were based on term coverage in relevant documents indicated by the user, and term frequencies in those documents and in the database. It would appear that on average the simpler the method for generating a context, the more likely it is to be of use in document ordering. Additionally, a context learner is most useful for initial queries of two or three words, which is in fact the most common case.

CONCLUSIONS

It would appear that there is scope for context learning to help IR systems adapt to users' particular queries, and that it is possible to 'learn' from one query to another, disregarding online session boundaries. However, the precise way in which the context is formed and used is critical and therefore applications of this idea to different systems must involve reassessment of the possible parameters and module combinations.

REFERENCES

1. Göker, A. *An investigation into the application of machine learning in information retrieval.* PhD thesis, Department of Information Science, City University, London, 1994.
2. Walker, S. *and* Hancock-Beaulieu, M. *Okapi at City: an evaluation facility for interactive information retrieval.* London: British Library, 1991. (British Library Research Report 6056)
3. Belkin, N.J. Interfaces for information retrieval systems, user modelling in information retrieval systems. *In: Proceedings of the European Summer School in Information Retrieval, Bressanone, 9–12 July 1990.* Italy: Associazone Italiana per l'informatica ed il calcolo automatico, 1990, 274–334.

(*Received September 1996*)

This article has been cited by:

1. Rajendra Prasath, Vijai Kumar, Sudeshna Sarkar. 2015. Assisting web document retrieval with topic identification in tourism domain. *Web Intelligence* **13**:1, 31-41. [CrossRef]
2. Seda Ozmutlu, Huseyin C. Ozmutlu, Gencer C. Cosar. 2011. Neural network applications for automatic new topic identification of FAST and Excite search engine transaction logs. *Expert Systems* **28**:2, 101-122. [CrossRef]
3. Seda Ozmutlu, Huseyin C. Ozmutlu, Buket Buyuk. 2008. A Monte-Carlo simulation application for automatic new topic identification of search engine transaction logs. *Simulation Modelling Practice and Theory* **16**:5, 519-538. [CrossRef]
4. H. Cenk Ozmutlu, Fatih Cavdur, Seda Ozmutlu. 2008. Cross-validation of neural network applications for automatic new topic identification. *Journal of the American Society for Information Science and Technology* **59**:3, 339-362. [CrossRef]
5. Ayşe Göker, Hans Myrhaug. 2008. Evaluation of a mobile information system in context. *Information Processing & Management* **44**:1, 39-65. [CrossRef]
6. Seda Ozmutlu. 2006. Automatic new topic identification using multiple linear regression. *Information Processing & Management* **42**:4, 934-950. [CrossRef]
7. H. Cenk Ozmutlu, Fatih Cavdur, Seda Ozmutlu. 2006. Automatic new topic identification in search engine transaction logs. *Internet Research* **16**:3, 323-338. [Abstract] [Full Text] [PDF]
8. H. Cenk Ozmutlu, Fatih Çavdur. 2005. Application of automatic topic identification on Excite Web search engine data logs. *Information Processing & Management* **41**:5, 1243-1262. [CrossRef]
9. Seda Özmutlu, Fatih Çavdur. 2005. Neural network applications for automatic new topic identification. *Online Information Review* **29**:1, 34-53. [Abstract] [Full Text] [PDF]
10. H. Cenk Özmutlu, Fatih Çavdur, Seda Özmutlu, Amanda Spink. 2004. Neural network applications for automatic new topic identification on excite web search engine data logs. *Proceedings of the American Society for Information Science and Technology* **41**:1, 310-316. [CrossRef]
11. Ayse Goker, Daqing He. 2003. Personalization via collaboration in Web retrieval systems: A context based approach. *Proceedings of the American Society for Information Science and Technology* **40**:1, 357-365. [CrossRef]
12. Susan Jones, Olga Vechtomova, Stephen Robertson. 2002. A tool for comparative evaluation in an interactive environment. *Journal of Information Science* **28**:6, 493-503. [CrossRef]
13. Daqing He, Ayşe Göker, David J Harper. 2002. Combining evidence for automatic Web session identification. *Information Processing & Management* **38**:5, 727-742. [CrossRef]

14. Kiduk Yang, Kelly L Maglaughlin, Gregory B Newby. 2001. Passage feedback with IRIS. *Information Processing & Management* **37**:3, 521-541. [CrossRef]